

# Generating Descriptions that Summarize Geospatial and Temporal Data

Martin Molina

Department of Artificial Intelligence  
Universidad Politécnica de Madrid, Spain  
martin.molina@upm.es

Amanda Stent

AT&T Labs – Research  
Florham Park, NJ, USA  
stent@research.att.com

## Abstract

*Effective data summarization methods that use AI techniques can help humans understand large sets of data. In this paper, we describe a knowledge-based method for automatically generating summaries of geospatial and temporal data, i.e. data with geographical and temporal references. The method is useful for summarizing data streams, such as GPS traces and traffic information, that are becoming more prevalent with the increasing use of sensors in computing devices. The method presented here is an initial architecture for our ongoing research in this domain. In this paper we describe the data representations we have designed for our method, our implementations of components to perform data abstraction and natural language generation. We also discuss evaluation results that show the ability of our method to generate certain types of geospatial and temporal descriptions.*

## 1. Introduction

In the field of Artificial Intelligence, systems have been developed that produce summaries over large data sets, to help humans interpret and analyze trends in the data. These systems usually operate in two stages: data abstraction (including qualitative interpretation), and summary generation (including natural language generation and multimedia presentation planning). For example, data-to-text systems have recently been developed that generate textual summaries of numeric and other non-linguistic data to help people understand weather forecasts [16], engineering data [24] or medical records [9]. Other systems have been developed that produce multimedia summaries (for example, in hydrology [13]). In general, these solutions share an abstract architecture, but require domain specific knowledge.

Summarization of geospatial and temporal data can be particularly useful. Data summarization in this area is nowadays especially interesting given the increasing availability of large amounts of GPS data recorded by a variety of different devices (mobile phones, cars, etc.). The summarization of geospatial and temporal data can be useful for many tasks including fleet monitoring, mission data analysis (military or scientific missions), planning trips, schedule verification, detection of behavior changes, intelligence analysis, transportation pattern analysis, etc. In addition, geospatial and temporal data summarization can be combined with domain-dependent summarization strategies to build more complex summarization systems (for example, systems that make trip recommendations, or that analyze social networks).

The goal of our research is to build general-purpose reusable tools for automatic generation of multimedia summaries of geospatial and temporal data. Although researchers have proposed separate solutions for summarization of temporal data and of geospatial data, there are currently no methods that integrate both dimensions in a practical way and that are able to generate multimedia presentations. Since geospatial and temporal data are present in many different types of tasks and we are interested in building general-purpose tools, we have initially focused on creating representations for the data used during summarization, and on designing a general, modular summarization method.

In this paper we focus on the generation of textual summaries (which are intended to accompany multimedia presentations). The work presented here corresponds to the implementation of an experimental method with evaluation results. It is part of the initial results of our ongoing research in this domain. We describe our representations for geospatial and temporal information, the representation for discourse structures of summaries, and the data abstraction and summary generation procedures we have implemented. We also present an evaluation and a comparison with related work.

## 2. The summarization task

The task we are addressing in this research is the automatic generation of multimedia presentations that summarize geospatial and temporal data (such as GPS traces or call records). This task involves several subtasks, including data abstraction, discourse planning, text generation, and multimedia presentation generation. In this paper we focus on the data abstraction, discourse planning and text generation tasks.

The input to the text summarizer is GPS traces corresponding to an individual (e.g. a person or a vehicle). These GPS traces include date/time, latitude and longitude sampled periodically over a certain interval of time (for example, a number of days, months or years). The output is a short natural language description (no more than 200 words) that summarizes the behavior of the individual in geospatial and temporal terms.

### **A trip to Costa Rica and Panama**

*The trip started on January 2<sup>nd</sup> 2003 and lasted for two months. San Jose was the initial and final place of the trip. During the first 5 weeks, we stayed in Puntarenas most of the time. The rest of the time, we were travelling to different places of Costa Rica and Panama.*

*During the time we were in Puntarenas, we visited some nearby places on the Pacific Coast. For example, we went every weekend to San Lucas during the morning and we went three nights to Bajamar. We also visited Tarcoles and Puerto Caldera some days during the morning.*

Figure 1. Partial example of a trip summary.

Figure 1 shows a representative example that illustrates the type of text summary that interests us. This example illustrates some characteristics of this type of summary:

- *Geospatial and temporal aggregation.* The summary includes geospatial and temporal aggregations that abstract away from specific details. For example, *Puntarenas* and *Costa Rica* are two abstractions of sets of latitude/longitude pairs, while *two months* and *the first 5 weeks* are abstractions over time intervals which are themselves abstractions over sets of time points in the input data. The summary is presented at a level of abstraction that is dynamically chosen (either according to length restrictions on the output summary, or by the user).
- *Discourses with progressively more detailed descriptions.* The summary is structured so that it proceeds from a high level of geospatial and temporal abstraction to more specific information. In the example, the headline summarizes the whole trip in one sentence. After the headline, the initial paragraph

gives a global description, while the next paragraph gives more details around a particular location.

- *Summaries of patterns.* Another characteristic of this type of summary is the description of temporal periods that summarize sets of events. For instance, the example in the figure includes the sentence *we went every weekend to San Lucas during the morning* which identifies a periodic visit to a particular place.

## 3. The method

Our summarization method consists of two main tasks (Figure 2): *geospatial and temporal abstraction* and *text planning*. In turn, text planning is carried out by three tasks, *discourse planning*, *sentence planning* and *surface realization*, that correspond to the process followed by most natural language generation systems [14] [15].

### 3.1. Geospatial and temporal abstraction

The goal of geospatial and temporal abstraction is to analyze the data in order to generate abstractions during the summarization process. Abstraction is used to identify the main statement of the summary. It is also used during discourse planning to produce supporting information for the main summary statement. Abstraction is performed by a set of prefixed *abstraction operators*. These operators can be classified into the following types:

- *Itinerary abstraction.* The input to itinerary abstraction is a GPS trace. We automatically preprocess this data to identify the nearest toponym for each pair <latitude, longitude>. We then group time points around each toponym to form time intervals. The output from itinerary abstraction is a time-ordered list of pairs <toponym, time-interval> describing the complete history of movement captured by the GPS trace. Itinerary abstraction uses web accessible databases (such as [www.geonames.org](http://www.geonames.org)) which provide automatic translation between geographical references and toponyms. Itinerary abstraction is parameterized, so that for example the initial geospatial granularity (city, neighborhood, etc.) can be specified.
- *Geospatial aggregation.* We abstract the itinerary at different levels of geospatial granularity (e.g. city, state). Also, we find the most specific geographic area that covers all the places in the itinerary. Itinerary abstraction uses a geographical database providing information based on a geographical ontology that relates toponyms of places (towns, cities, lakes, rivers, etc.) and spatial areas (regions, states, nations, etc.).

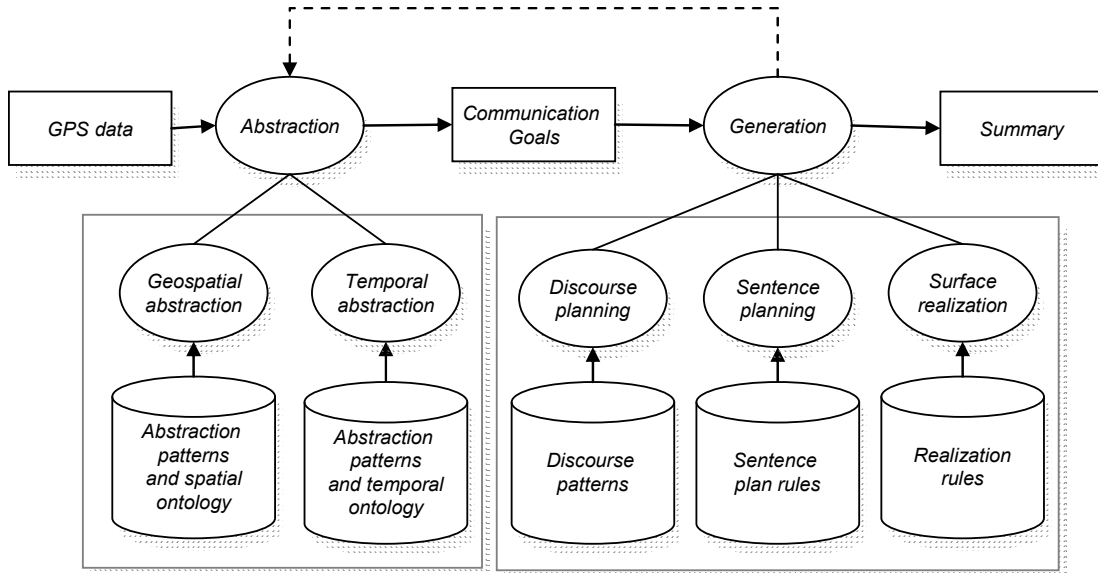


Figure 2. Main components of our method for automatic summarization of geospatial and temporal data. In the figure, rectangles represent input-output data, ellipses represent tasks or subtasks and cylinders represent knowledge bases.

- *Temporal aggregation.* We aggregate over the time intervals corresponding to the different levels of geospatial granularity. To each place at each level of abstraction, we can automatically assign information such as the total duration of the itinerary in this place and, when the place has been visited more than once, the periodicity or frequency of the visits. For example, temporal abstraction can generate a logical representation corresponding to descriptions such as *we went to Jersey City twice a month* or *he visited Madrid every summer*. Temporal abstraction uses a knowledge base that includes information about time and calendars.

To find periodic temporal patterns for a given place, we combine two abstraction procedures. The first procedure follows a bottom-up temporal aggregation search strategy directed by prefixed time periods. It starts at the level of hours, then days, weeks and so on. At every level it looks for repetitions of time gaps between consecutive visits. This procedure generates descriptions such as *3 hours every 2 days*.

The second procedure analyzes the repetition of calendar dates. For this purpose, a temporal ontology with calendar attributes is used. Examples of these attributes are: *semester* (with values *first, second*), *season* (*winter, spring, summer, autumn*), *part of month* (*first, middle, end*), *part of week* (*weekday, weekend*), *holiday* (*christmas day, ...*). For every time interval in a given place, the procedure generates the values of the calendar attributes. Then, discriminative values of attributes are selected, i.e., attributes with the same value in a high percentage  $H$  of intervals (for example,  $H > 85\%$ ). Since calendar attributes are not independent, a prefixed priority

scheme is followed, starting with fine-grained attributes. Finally, the procedure quantifies the dates to get qualitative descriptions such as *every summer* or *many weekends*.

The output from geospatial and temporal abstraction is a multi-level description of an abstract itinerary. We use an event-based representation with a tuple for each event  $\langle \text{identification}, \text{space}, \text{time} \rangle$  where *identification* corresponds to information that identifies the event (number of event, subject, verb, etc.), *space* is a predicate that describes a spatial property and *time* is a predicate that describes a temporal property. Figure 3 shows examples of predicates that we use for space and time. This representation borrows heavily from other spatio-temporal representations used in text analysis such as SpatialML [10] and TIDES [5]. For example, to represent the temporal periodicity of events we use the predicate **periodic-times**(*quantity, unit, period, date*) where *quantity* is the number of times, *unit* is the time unit, *period* is the period and *date* is a calendar date (e.g. for calendar date we follow the extension of the ISO 8601 standard defined by TIDES [5]). The predicate:

**periodic-times**(1, D, P2W, XXXX-XX-WE)

represents *one day every two weeks on the weekend*. In this example, the first argument with value 1 is quantity, the second with value D is the representation for the day unit (W week, M month, Y year, etc.), the third argument corresponds to the duration of 2 weeks and the fourth is a representation of a general date weekend.

| Type  | Predicates   |
|-------|--|
| Time  | <b>Format:</b><br>date( <i>date</i> )<br>time-of-day( <i>time</i> )<br>duration( <i>duration</i> )<br>periodic-times( <i>quantity</i> , <i>unit</i> , <i>period</i> , <i>date</i> )<br>number-of-times( <i>quantity</i> , <i>unit</i> )<br>number-of-dates( <i>quantity</i> , <i>date</i> )<br>... |
|       | <b>Examples:</b><br>date(2009-06-01)<br>time-of-day(T18:30)<br>duration(P20D)<br>periodic-times(1, D, P2W, XXXX-XX-WE)<br>number-of-times(2, W)<br>number-of-dates(all, XXXX-12-25)  |
| Space | <b>Format:</b><br>geo-coordinates( <i>latitude</i> , <i>longitude</i> )<br>toponym( <i>name</i> )<br>place-granularity( <i>granularity</i> )<br>distance( <i>distance</i> , <i>unit</i> )<br>direction( <i>direction</i> )<br>spatial-relation( <i>relation</i> )<br>...                           |
|       | <b>Examples</b><br>geo-coordinates(40.45, 73.59)<br>toponym(Madrid)<br>place-granularity(country)<br>distance(50, mi)<br>direction(WS)<br>spatial-relation(near)   |

Figure 3. Examples of predicates for temporal and spatial descriptions.

### 3.2. The discourse planner

The output from geospatial and temporal abstraction includes a top-level, single-assertion communication goal that summarizes the entire data set. The discourse planner selects elements from the abstract itinerary description that provide supporting information for the top-level goal. As we have mentioned, the discourse follows a progressive description at different levels of abstraction (from more general to more specific) with explicit presentation of facts in the form of list of facts, chronological descriptions, representative examples, etc.

The representation used by the discourse planner is Rhetorical Structure Theory (RST) [11]. The discourse planner has a knowledge base of discourse patterns (RST tree templates) for discourse segments. The set of discourse patterns is implemented as the knowledge base of a planner that uses a modified version of HTN planning (Hierarchical Task Network) [7]. One of the advantages of this approach is that the knowledge base could be extended to generate not only text but also multimedia presentations (as is done e.g. in [13]).

According to the terminology of HTN planning, the model includes planning-tasks, planning-methods, preconditions and operators. In our method, each planning-task corresponds to a communication goal (e.g.,

to inform about the details of a relevant event) and each planning-method corresponds to a discourse pattern. A pattern has a set of communication sub-goals that corresponds to the rhetorical relations. Thus, the whole discourse strategy comprises several levels of communication-goals and candidate discourse patterns.

|             |  |
|-------------|--|
| GOAL:       | elaborate_trip( <i>x</i> )   |
| PATTERN:    | chronological_elaboration( <i>x</i> , <i>y</i> )   |
| CONDITIONS: | trip( <i>x</i> ) $\wedge$ spatial_granularity( <i>y</i> ) $\wedge$<br>$\wedge$ number_of_visits( <i>x</i> , <i>y</i> , <i>n</i> ) $\wedge$ ( <i>n</i> < 5) |

Figure 4. Example of relation between goal and discourse.

Figure 4 shows an example that relates a goal with a discourse pattern. In general, several different patterns can be associated with the same goal. In this example, the goal is to elaborate the information of a trip. The discourse pattern associated with this goal is called *chronological elaboration*. To select this pattern, the condition is that the trip must have just a few numbers of visited places (less than five). The visited places during a trip can be described at different levels of spatial granularity (e.g. nation, state, town, address, etc.). In the condition, the specific level of spatial granularity is identified by the predicate **spatial\_granularity**(*y*). Figure 5 shows the description of the discourse pattern **chronological-elaboration**(*x*,*y*). The pattern includes a list of subgoals. Each subgoal is identified with the predicate **goal**(*g*, *p*), where *g* is the presentation goal and *p* is a variable that identifies the content of the presentation produced by the presentation goal. The discourse pattern is associated with a RST relation (in the example, the *elaboration* relation) and the corresponding variables associated to the presentation goals establish the roles for nucleus and satellite.

|             |  |
|-------------|--|
| PATTERN:    | chronological_elaboration( <i>x</i> , <i>y</i> )   |
| SUBGOALS:   | {goal(inform( <i>z</i> ), <i>p</i> <sub>1</sub> ), goal(list_chronological( <i>u</i> ), <i>p</i> <sub>2</sub> )}   |
| DISCOURSE:  | relation(elaboration, nucleus({ <i>p</i> <sub>1</sub> }), satellite({ <i>p</i> <sub>2</sub> }))  |
| CONDITIONS: | trip( <i>x</i> ) $\wedge$ spatial_granularity( <i>y</i> ) $\wedge$<br>$\wedge$ global_place( <i>x</i> , <i>z</i> ) $\wedge$ chronological_places( <i>x</i> , <i>y</i> , <i>u</i> ) |

Figure 5. Example of discourse pattern.

As the previous examples show, the conditions may include predicates that correspond to calls to abstraction operators (for example, **number\_of\_visits**(*x*,*y*,*n*) or **chronological\_places**(*x*,*y*,*u*)). These operators are provided by the geospatio-temporal abstraction component and are invoked backwards when the discourse planner selects a candidate goal or a candidate pattern. To construct the discourse plan the planner repeatedly (a) selects a candidate discourse pattern for the current goal, (b) gathers additional abstracted information according to the conditions of the pattern, and (c) refines the communication goal with the sub-goals.

As a result of this process a discourse plan is generated. It is represented as a tree of nodes. Each internal node is a RST relation, while the leaves are propositions. Propositional content is captured using a flat predicate-argument structure representation. Figure 6 shows a partial example of a discourse plan with this representation, corresponding to the summary:

*We went to NYC five weekends. For example, we went there on Saturday the 14th during the morning and on Saturday the 21st during the afternoon.*

```
objects(..., object(o8, NYC), ...)
...
node(n2,
  proposition(statement,
    {event(e), a0(we), pred(e, go),
      place(e,object(o8)), time(e,t),
      number-of-dates(t, 5, XXXX-XX-WE)}))
node(n3,
  proposition(statement,
    {event(e), a0(we), pred(e, go),
      place(e,object(o8)), time(e,t),
      date(t,XXXX-14-WXX-06TMO)}))
node(n4,
  proposition(statement,
    {event(e), a0(we), pred(e, go),
      place(e,object(o8)), time(e,t),
      date(t,XXXX-21-WXX-06TAF)}))
node(n8,
  relation(conjunction,
    nucleus([n3,n4]), satellite([])))
node(n9,
  relation(exemplify,
    nucleus([n2]),satellite([n8])))
```

Figure 6. Partial example of a discourse plan.

### 3.3. The sentence planner

For the sentence planner we use a modified version of the publicly available SPaRKY sentence planner [21]. SPaRKY takes as input a discourse plan (a tree with rhetorical relations on the internal nodes and a proposition representing a text span on each leaf), and outputs one or more sentence plans (each a tree with discourse cues and/or punctuation on the internal nodes). SPaRKY is a two-stage sentence planner. First, possible sentence plans are constructed through a sequence of decisions made using only local information about single nodes in the discourse plan. Second, the possible sentence plans are ranked using a user or domain-specific sentence plan ranker that evaluates the global quality of each sentence plan [23]. Sentence plan construction in SPaRKY involves three tasks: span ordering, sentence aggregation (deciding whether to realize a pair of propositions as a single clause, a single sentence, or two sentences), and discourse cue selection. In our version of SPaRKY, for sentence plan construction we use sentence planning rules automatically extracted from the RST-DT

corpus [1] as well as some hand-written rules (details about the extraction procedure can be read in [20]). For sentence plan ranking we use an n-gram language model trained on the English Gigaword corpus [8]. Examples of sentence plan rules are shown in Figure 7.

### 3.4 The surface realizer

For the surface realizer we currently use SimpleNLG [6]. SimpleNLG is a rule-based surface realizer. We perform a straightforward transformation of our flat predicate-argument representation for propositions into SimpleNLG input. When the sentence planner decides that two or more propositions should be aggregated into a single clause, we merge co-referential arguments and identical predicates in the input propositions. When the sentence planner decides that two or more propositions should be aggregated into a single sentence (separated by a comma or semi-colon), we realize each proposition separately and then insert the desired punctuation.

```
Rule temporal-before
Conditions:
  Child 0 type nucleus, Child 1 type satellite
  Children are not coreferential
  Children have same verb
  Children temporal order is Child 1 Child 0
Effects:
  Order Child 0 Child 1
  Cues Child 1 before
  Punc Child 0 comma
Frequency: 6

Rule elaboration-gen-specific
Conditions:
  Child 0 type nucleus, Child 1 type satellite
  Children are not coreferential
  Children do not have same verb
  Shortest child is Child 0
Effects:
  Order Child 0 Child 1
  Punc Child 0 comma
Frequency: 2
```

Figure 7. Glosses of example sentence plan rules.

## 4. Evaluation

We have created an initial implementation of the method presented above and evaluated using data from the Monroe corpus [19]. In this section we briefly describe our implementation before presenting our evaluation results.

Our data abstraction module and the discourse planner are implemented in Prolog. We have implementations of both the inference procedures and the corresponding knowledge bases (temporal and geospatial ontologies and discourse patterns). The sentence planner and surface realizer are implemented in Java; the rules for the surface realizer are stored in XML.



## A 21-day trip to the Andean Community

From Madrid, we traveled to the Andean Community and visited 5 cities of different countries during 21 days. From Lima, we traveled to La Paz and stayed there for 4 days. Next, we visited Quito for 4 days, Caracas for 3 days and Bogota for 3 days. Finally we went back to Lima where we spent other 3 days before we went back to Madrid on December 20. [More >>](#)



Figure 8. Example of global presentation of geographical and temporal information with text summaries and graphical tools.

Our initial implementation also includes the design of multimedia presentation module that provides context for the text summaries. Figure 8 shows an example of our design for a multimedia composition that presents summaries about trips. In the figure, on the left hand side, a text summary describes the global trip. On the right hand side, geospatial and temporal information is presented using interactive graphics. The geographical presentation includes description points for specific places on the map; when the user clicks on one he or she is presented with a specific summary. The overall presentation also includes a temporal bar at the bottom of the map, and the user can go forward and backward in time using the left and right arrows at the bottom of the map. The temporal bar is implemented using the *Timeline* tool from the *SIMILE Project* [12].

We are currently conducting black-box and glass-box evaluations of our summarization method. For example, we carried out an evaluation of the coverage and discriminative power of our sentence planning rules, which is described in [20]. Here we present an evaluation to test the quality of the generated natural language summaries. We applied a procedure based on the comparison of (a) original text descriptions about trips written by humans in natural language and (b) text descriptions in natural language generated automatically

by our system. For this purpose we first selected a set of 20 descriptions about trips from the plan summaries produced by participants in the Monroe dialog collection [19]. Each example describes one or more trips made by a single vehicle in and around Rochester, NY. For each example, the corresponding raw data was manually generated (for the sake of simplicity, we used as input data time-ordered list of pairs <toponym, time-interval> instead of raw GPS data, so the evaluation did not include itinerary abstraction but it did include geospatial and temporal aggregation as well as summary generation). The input data was processed by our system to automatically generate new text descriptions. We then compared the automatically generated summaries with two human-authored summaries for each set of raw data. The average length of the automatically generated summaries was 45.1 words (with 3 punctuation marks indicating clause boundaries), as compared with 49.1 words (and 4.1 punctuation marks) for summaries generated by one human and 47.7 words (and 3.25 punctuation marks) for summaries generated by the other.

The comparison for each pair covered the following features: (1) correct, i.e., the generated text does not contradict the original text, (2) complete, i.e. the generated text covers everything that was said in the original text, (3) grammatical and natural, i.e, the

formulation of the generated text is fluent and idiomatic English, (4) equivalent, i.e., the generated text is equal to or better than the original description (no word would be changed). We rated each summary a 1 for naturalness and grammatical if both title and body were natural and grammatical; a .5 if either the title or the body was ungrammatical or non-idiomatic, and a 0 if neither were grammatical or natural.

Our automatically generated summaries obtained an average rating of 1.0 for correctness, 0.95 for completeness, 0.56 for naturalness and grammaticality, and 0.25 for equivalence. The evaluation showed that the method generates acceptable text descriptions, especially for certain spatial aggregations and chronological descriptions of trips. There remain several issues, including:

- The titles are not always natural (see 2<sup>nd</sup> example below). This is partly a failing of the representation for the input to the sentence planner, and partly that we need to perform a deeper analysis of the structure of headlines in human-authored texts.
- The sentence planner is very conservative in choosing to use pronouns (see 2<sup>nd</sup> example below). If the sentence planner can do referring expression generation after sentence planning, it can use more pronouns while still producing correct output.
- Especially in lists, but also occasionally at other times, facts are ordered incorrectly or discourse cues are used inappropriately (see 3<sup>rd</sup> example below). We can easily create a small number of hand-written rules to correct these errors, which are due to data sparsity in the training data for the sentence planner.

|  |
|--|
| <p><b>Trip to Highland Avenue for 2 hours</b></p> <p><i>The second electric crew went to Highland Avenue for 2 hours.</i></p>  |
| <p><b>Make 2 trips in Rochester</b></p> <p><i>The origin of every trip was at Strong Memorial Hospital. The second ambulance made 2 trips on February 1 2007 in Rochester. The second ambulance went at 12 o'clock to the intersection of NY252 and NY383 for 15 minutes. Then it went 2 places in Rochester at 13 o'clock for 1 hour.</i></p>   |
| <p><b>Visit 4 places in Rochester</b></p> <p><i>The first road crew visited 4 places in Rochester for 12 hours. First the first road crew visited the intersection of Culver Road and East Avenue for 3 hours. Second the first road crew visited I490 at the Inner Loop for 3 hours. Third it visited Greater Rochester Airport for 3 hours. In this case it visited the intersection of NY386 and NY104 for 3 hours.</i></p> |

Figure 10. Examples of texts automatically generated for the Monroe Corpus.

In addition to this, we also performed a number of comparison tests with other more complex examples based on texts found in public web sites. In general, the method generated correct texts, and periodic patterns were adequately described. The main limitation of our

method is that it is currently not able to present richer geographical descriptions about locations based on distances, orientations, etc., or to take into account the knowledge of the user about the geographical area. An example location description our system is not currently capable of planning or producing is *Bajamar, which is 25 miles from Puntarenas*. As part of our future work we plan to extend our method with more detailed spatial knowledge so that we can generate these types of location descriptions.

## 5. Related work

The work that we present in this paper is related to abstraction methods and data summarization systems. In the area of temporal abstraction, different methods have been proposed; for example, methods for reasoning about temporal granularities with algebraic or logical approaches [3] or methods that follow a knowledge-based approach for temporal abstraction [2] [18]. Our method takes a knowledge based approach but it uses a different representation for domain knowledge (events, abstraction operators, patterns, etc.) that considers not only the temporal dimension but also geospatial references.

Our method for data summarization has similar abstract components to other data-to-text systems (for weather forecasting [16], engineering [24] or medicine [9]). However, our method was designed for summarization of geospatial and temporal data with the purpose of being combined with other types of domain-specific data. This means that it can be reused (wholly or partially) and combined with domain-specific data summarization methods, for example in traffic pattern observation, fraud detection and intelligence analysis.

There are other systems that perform data summarization of geographical data such as the Coral system [4] and the RoadSafe system [22]. The main difference between these systems and our method is the type of presentation and the type of data handled. Coral gives travel directions (based on spatial data, but without temporal references) and RoadSafe explains how weather conditions affect certain spatial areas (without considering historical behavior). By contrast, our system provides general-purpose summaries along both geospatial and temporal dimensions. On the other hand, Coral generates certain spatial descriptions related to paths (such as *follow the Main Street*) that the current version of our method is not able to generate.

## 6. Conclusions

In this paper we have presented a method for generating textual summaries of geospatial and temporal data. Our method shares a common architecture with

other data summarization systems. However, it includes several novel aspects: (1) to our knowledge it is the first application with the goal of summarizing geospatial and temporal information as it is provided by GPS data, (2) the abstraction method has been designed specifically for this task integrating both geographical and temporal knowledge, and (3) we specifically constructed knowledge bases for natural language generation (following semi-automatic procedures) for spatio-temporal descriptions that can be reused for different applications.

An evaluation of our method shows that we can generate adequate descriptions of recurrent spatio-temporal patterns and event sequences. In the future we will extend the method to include user modeling (so that we can generate spatial descriptions based on the user's familiarity with the area, for example). We will also extend the method to generate summaries about multiple individuals (e.g. a fleet of vehicles, a collection of travelers).

## Acknowledgements

This work was partially supported by the Ministry of Science and Innovation of Spain within the VIOMATICA project (TIN2008-05837/TIN). The authors thank Enrique Parodi for the development of the multimedia presentation prototype that provides context for the text summaries.

## References

- [1] L. Carlson, D. Marcu, and M. E. Okurowski, "Building a discourse-tagged corpus in the framework of rhetorical structure theory", in *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, 2002.
- [2] S. Chakravarty and Y. Shahar, "A Constraint-based specification of periodic patterns in time-oriented data", in *Proceedings of the Sixth International Workshop on Temporal Representation and Reasoning (TIME-99)*, 1999.
- [3] C. Combi, M. Franceschet, and A. Peron, "Representing and reasoning about temporal granularities", *Journal of Logic and Computation*, vol. 14, no. 1, pp. 51-77, 2004.
- [4] R. Dale, S. Geldof, and J-P. Prost, "Using natural language generation in automatic route description", *Journal of Research and Practice in Information Technology*, vol. 37, no. 1, pp. 89-105, 2005.
- [5] L. Ferro, L. Gerber, I. Mani, B. Sundheim, and Wilson G., "TIDES 2005 Standard for the Annotation of Temporal Expressions", MITRE, 2005. Available from: [http://fofoca.mitre.org/annotation\\_guidelines/timex2\\_annot\\_guidelines.html](http://fofoca.mitre.org/annotation_guidelines/timex2_annot_guidelines.html).
- [6] A. Gatt and E. Reiter, "SimpleNLG: A realisation engine for practical applications", *Proceedings of ENLG-2009*.
- [7] M. Ghallab D. Nau, and P. Traverso, *Automated Planning: Theory and Practice*, Morgan Kaufmann, 2004.
- [8] D. Graff, "English Gigaword", Linguistic Data Consortium Catalog No. LDC2003T05, 2003. Available from <http://www.ldc.upenn.edu/>.
- [9] J. Hunter, A. Gatt, F. Portet, E. Reiter, and S. Sripada, "Using natural language generation technology to improve information flows in intensive care units", in *Proceedings of the 5th Conference on Prestigious Applications of Intelligent Systems*, 2008.
- [10] I. Mani, J. Hitzeman, J. Richer, D. Harris, R. Quimby, and B. Wellner, "SpatialML: Annotation Scheme, corpora, and tools", in *Proceedings of the Sixth International Language Resources and Evaluation Conference (LREC '08)*, 2008.
- [11] W. Mann and S. Thompson, "Rhetorical Structure Theory: Toward a functional theory of text organization", *Text*, vol. 8, no. 3, pp. 243-281, 1988.
- [12] S. Mazzocchi, S. Garland, and R. Lee, "SIMILE: Practical Metadata for the Semantic Web", *xml.com*, January 26, 2005.
- [13] M. Molina and V. Flores, "A presentation model for multimedia summaries of behavior", in *Proceedings of the International Conference on Intelligent User Interfaces*, 2008.
- [14] O. Rambow, S. Bangalore, and M.A. Walker, "Natural language generation in dialog systems", in *Proceedings of the Human Language Technology Conference*, 2001.
- [15] E. Reiter and R. Dale, *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge, UK, 2000.
- [16] E. Reiter, S. Sripada, J. Hunter, J. Yu, and I. Davy, "Choosing words in computer-generated weather forecasts", *Artificial Intelligence*, vol. 67, no. 1-2, pp. 137-169, 2005.
- [17] A. Setzer and G. Katz, "TimeML: Robust specification of event and temporal expressions in text", in *Proceedings of the Fifth International Workshop on Computational Semantics*, 2003.
- [18] Y. Shahar, "A framework for knowledge-based temporal abstraction", *Artificial Intelligence*, vol. 90, no. 1-2, pp. 79-133, 1997.
- [19] A. Stent, "The Monroe corpus", technical Report 728, Department of Computer Science, University of Rochester, March 2000.
- [20] A. Stent and M. Molina, "Evaluating automatic extraction of rules for sentence plan construction", in *Proceedings of the SIGdial Conference on Discourse and Dialogue*, 2009.
- [21] A. Stent, R. Prasad, and M.A. Walker, "Trainable sentence planning for complex information presentations in spoken dialog systems", in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2004.
- [22] R. Turner, Y. Sripada, and E. Reiter, "Generating approximate geographic descriptions", in *Proceedings of the European Natural Language Generation Conference (ENLG)*, 2009.
- [23] M. A. Walker, A. Stent, F. Mairesse, and R. Prasad, "Individual and domain adaptation in sentence planning for dialogue", *Journal of Artificial Intelligence Research*, vol. 30, pp. 413-456, 2007.
- [24] J. Yu, E. Reiter, J. Hunter, and C. Mellish, "Choosing the content of textual summaries of large time-series data sets", *Natural Language Engineering*, vol. 13, no. 1, pp. 25-49, 2007.